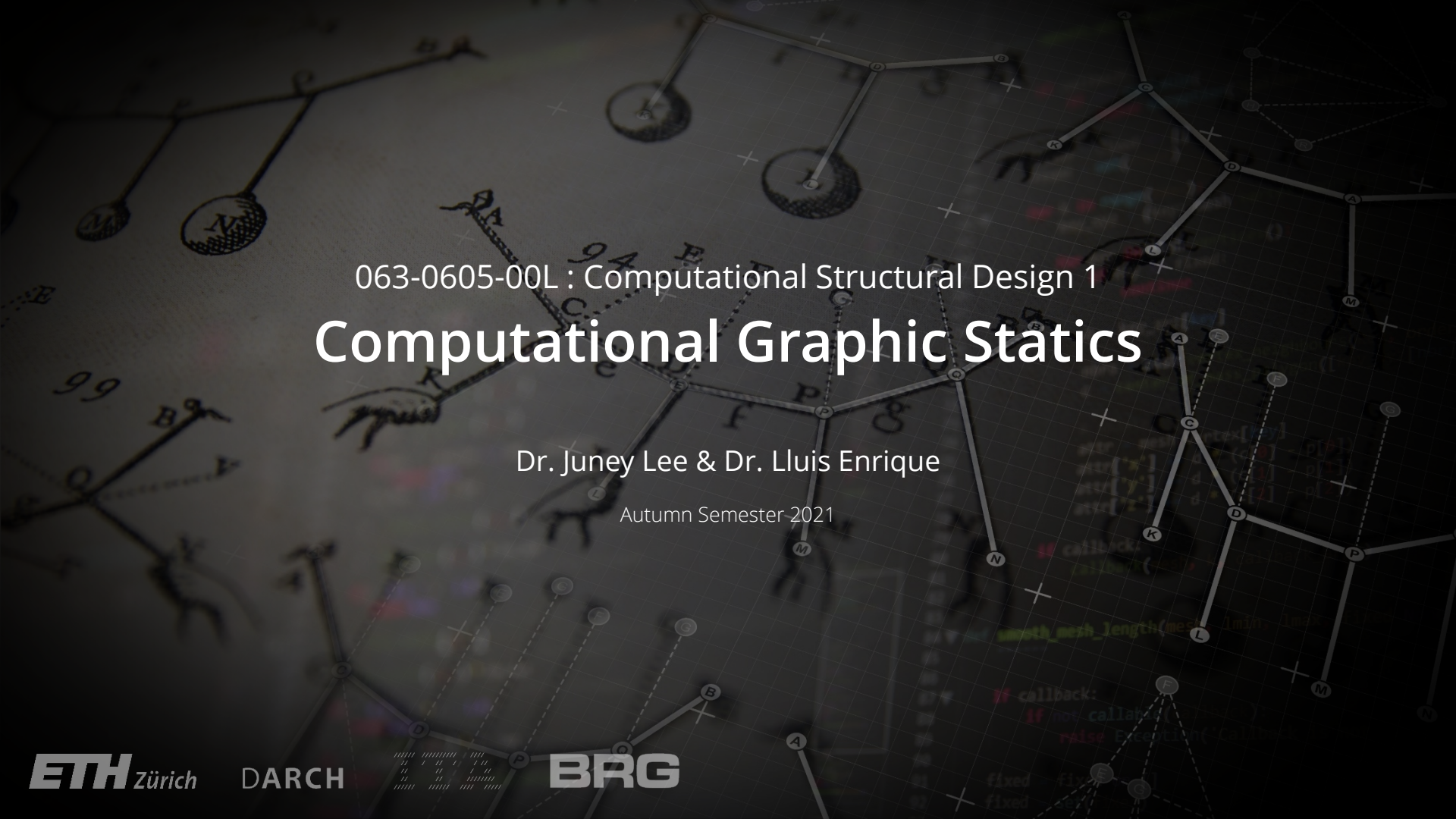


063-0605-00L : Computational Structural Design 1

Computational Graphic Statics

Dr. Juney Lee & Dr. Lluis Enrique

Autumn Semester 2021

ETH Zürich DARCH BRG

Week 2 Friday, October 1st

Hour 1	**9:45 – 10:30**	· Recap of Exercise 1 · **Lecture 2: Algorithmic design & thinking**
	10:30 – 10:45	· 15 min break
Hour 2	**10:45 – 11:30**	· **Tutorial 2: Single node bridge in GH**
	11:30 – 11:45	· 15 min break
Hour 3	**11:45 – 12:30**	· **Tutorial 2: Single node bridge in GH (cont'd)**

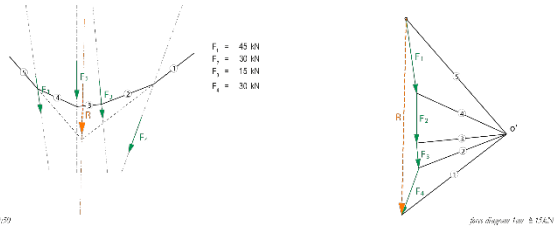
Task 1 Resultant of two non-parallel forces

Find the position, direction and magnitude of the resultant in the form and force diagram with the help of vector addition.

form diagram 1:50 *force diagram 1cm ≙ 15kN*

Task 2 Resultant of several non-parallel forces

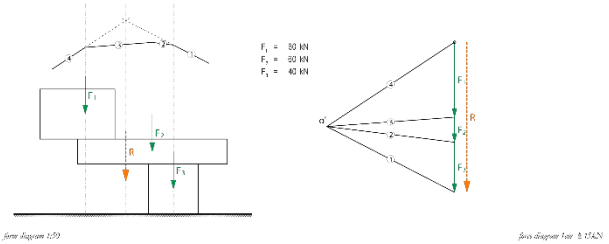
Find the position, direction and magnitude of the resultant in the form and force diagram with the help of a funicular polygon.



form diagram 1:50 *force diagram 1cm ≙ 15kN*

Task 3 Resultant of several parallel forces

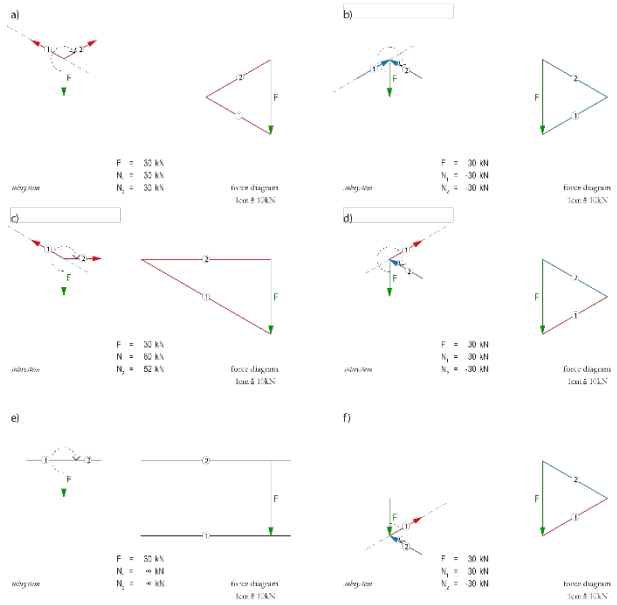
Find the position, direction and magnitude of the resultant in the form and force diagram with the help of a funicular polygon.



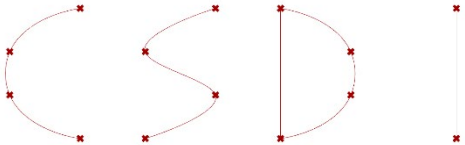
form diagram 1:50 *force diagram 1cm ≙ 15kN*

Task 4 Drawing subsystems

Draw a corresponding force diagram for each subsystem (a-f). Determine the magnitude [kN] of each force and mark its direction in the subsystem. Indicate tension forces with red and compression forces with blue. Explain the solution of situation c).

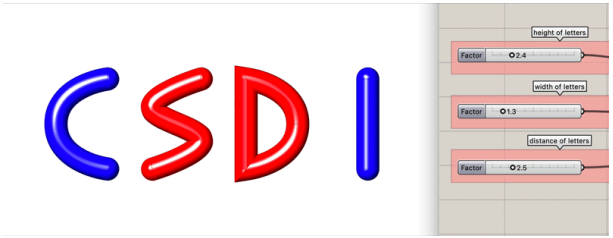


1. Point + line/curve construction

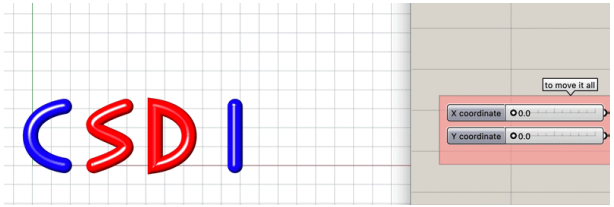


2. Coloured balloons

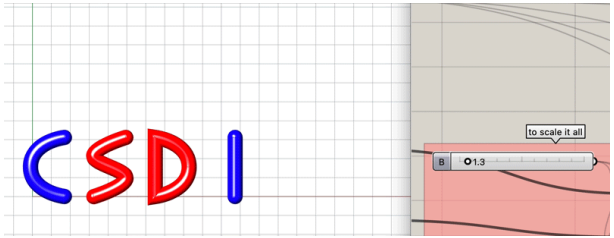
3. Height, width, spacing



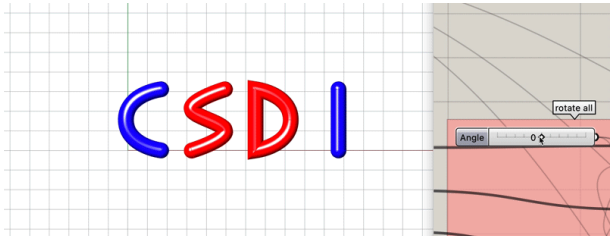
4. Position



5. Scale



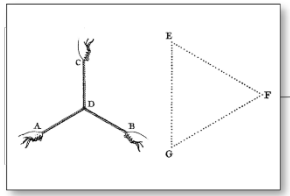
6. Rotation



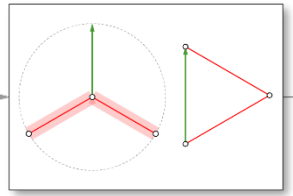
Course schedule

Structural Typology		Topic	Module	Week	hr		Topic	Instructor
1. planar structures	cables & arches	Introduction	I	Week 1 (9/24)	1	**Lecture 1**	**Introduction to graphic statics**	**Dr. Juney Lee**
					2	**Tutorial 1**	Basics of graphic statics on "blackboard"	Dr. Lluis Enrique
							Introduction to Rhino + Grasshopper (GH)	Lotte Aldinger
					3	**Work session 1**	***Exercise 1**: Reciprocal diagrams in Rhino + GH basics*	Instructors
		2D GS (procedural)	II				Quick recap of Exercise 1	Lotte Aldinger
				Week 2 (10/1)	1	**Lecture 2**	**Algorithmic design & thinking**	**Dr. Juney Lee**
					2	**Tutorial 2**	Single node bridge in GH (form to force)	Lotte Aldinger
					3			
				Week 3 (10/8)	2			
					3			
			III				Quick recap of Exercise 2	Lotte Aldinger
				Week 4 (10/15)	1	**Lecture 3**	**Computational graphic statics**	**Dr. Juney Lee**
					2	**Tutorial 3**	Multi-node bridge in Grasshopper	Dr. Lluis Enrique
					3			
				Week 5 (10/22)	1	**Work session 3**	***Exercise 3**: Multi-node bridge with GH*	Dr. Lluis Enrique + instructors
					2			
					3			
				Week 6	seminar week			
	trusses	2D GS (AGS)	IV				Quick recap of Exercise 3	Dr. Lluis Enrique
				Week 7 (11/5)	1	**Lecture 4**	**Algebraic graph statics (AGS)**	**Dr. Juney Lee**
					2	**Tutorial 4**	compas_ags + IGS (Interactive Graphic Statics) plugin	Ricardo Avelino
					3			
				Week 8 (11/12)	1	**Work session 4**	***Exercise 4**: Truss problems with IGS*	Ricardo Avelino + instructors
					2			
					3			
2. surface structures	shells	2.5 GS (TNA)	V				Quick recap of Exercise 4	Ricardo Avelino
				Week 9 (11/19)	1	**Lecture 5**	**Thrust Network Analysis (TNA)**	**Dr. Juney Lee**
					2	**Tutorial 5**	compas_tna + RV2 (RhinoVAULT 2) plugin	Dr. Juney Lee
					3			
				Week 10 (11/26)	1	**Work session 5**	***Exercise 5**: Shell design exercises with RV2*	Dr. Juney Lee + instructors
					2			
					3			
3. spatial structures	polyhedral structures	3D GS (polyhedral)	VI				Quick recap of Exercise 5	Dr. Juney Lee
				Week 11 (12/3)	1	**Lecture 6**	**3D graphic statics (3GS)**	**Dr. Juney Lee**
					2	**Tutorial 6**	compas_3gs + 3GS (3D graphic statics) plugin	Dr. Juney Lee
					3	**Work session 6**	***Exercise** 6: Spatial structures with 3GS*	
							Quick recap of Exercise 6	Dr. Juney Lee
				Week 12 (12/10)	1	**Lecture 7**	**Pt I. Outlook on computational graphic statics**	**Dr. Juney Lee**
							Pt II. Guest Lecture	**TBD**
					2	Q/A	Discussions, feedback, evaluations, etc.	

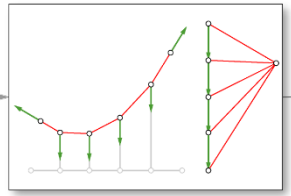
2D **2.5D** **3D**



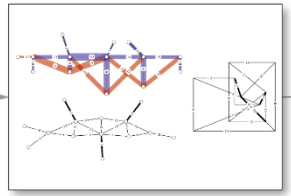
Module I
Basics of Graphic Statics



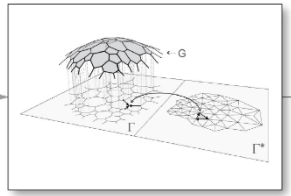
Module II
Interactive Graphic Statics I (single node)



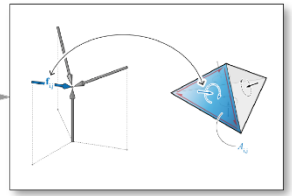
Module III
Interactive Graphic Statics 2 (multi node)



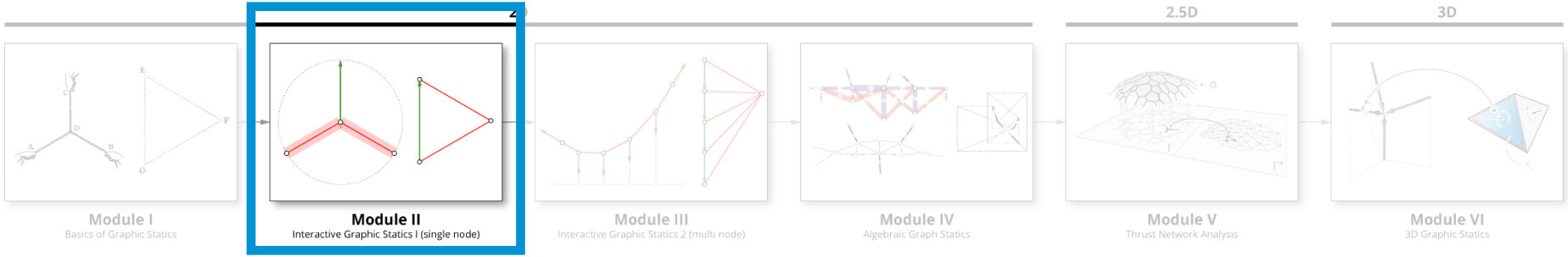
Module IV
Algebraic Graph Statics



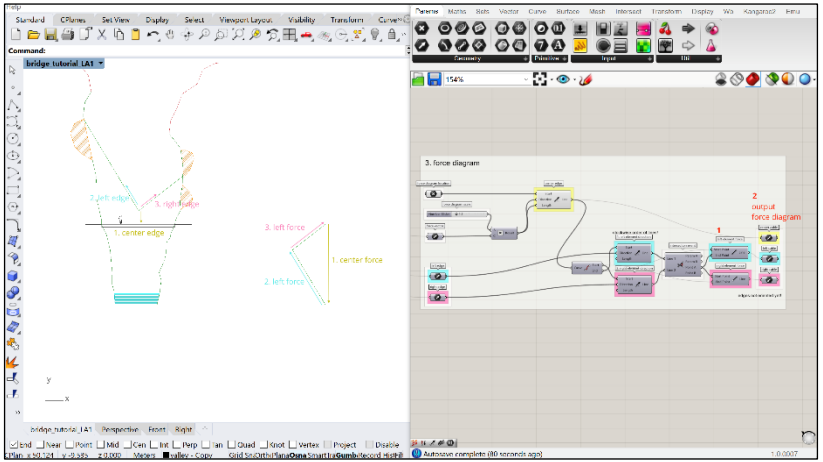
Module V
Thrust Network Analysis



Module VI
3D Graphic Statics



Module II · 2D procedural graphic statics 1



Windows
Rhinoceros

+

063-0605-00L : Computational Structural Design 1

Computational Graphic Statics

Lecture 2

Algorithmic design & thinking

Friday, October 1st, 2021

Dr. Juney Lee

ETH Zürich DARCH BRG

Computational Graphic Statics

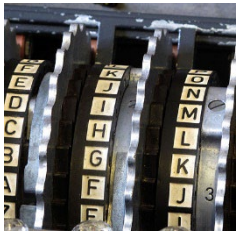
Universal Computing “Turing” Machine

… a machine capable of computing anything that is computable by executing instructions (program) stored on tape (memory), allowing the machine to be programmable …

“On Computable Numbers” | Alan Turing (1936)

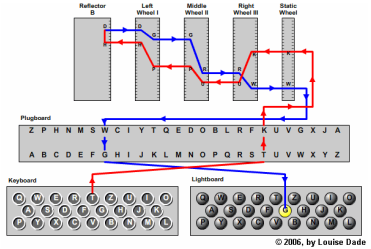
The Imitation Game (2014)

ATTACKATDAWN



Encryption machine (“Enigma”)

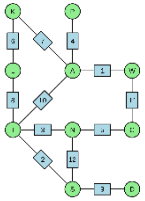
”Scrambler” using rotor configurations



Encryption key

Algorithm

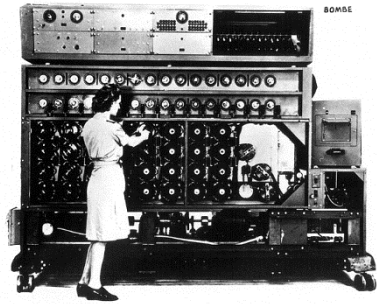
WSNPNLKLSTCS



Ciphertext	W	S	N	P	N	L	K	L	S	T	C	S
Plaintext "crib"	A	T	T	A	C	K	A	T	D	A	W	N
Message position	1	2	3	4	5	6	7	8	9	10	11	12
Upper drum setting	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
Middle drum setting	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
Lower drum setting	A	B	C	D	E	F	G	H	I	J	K	L

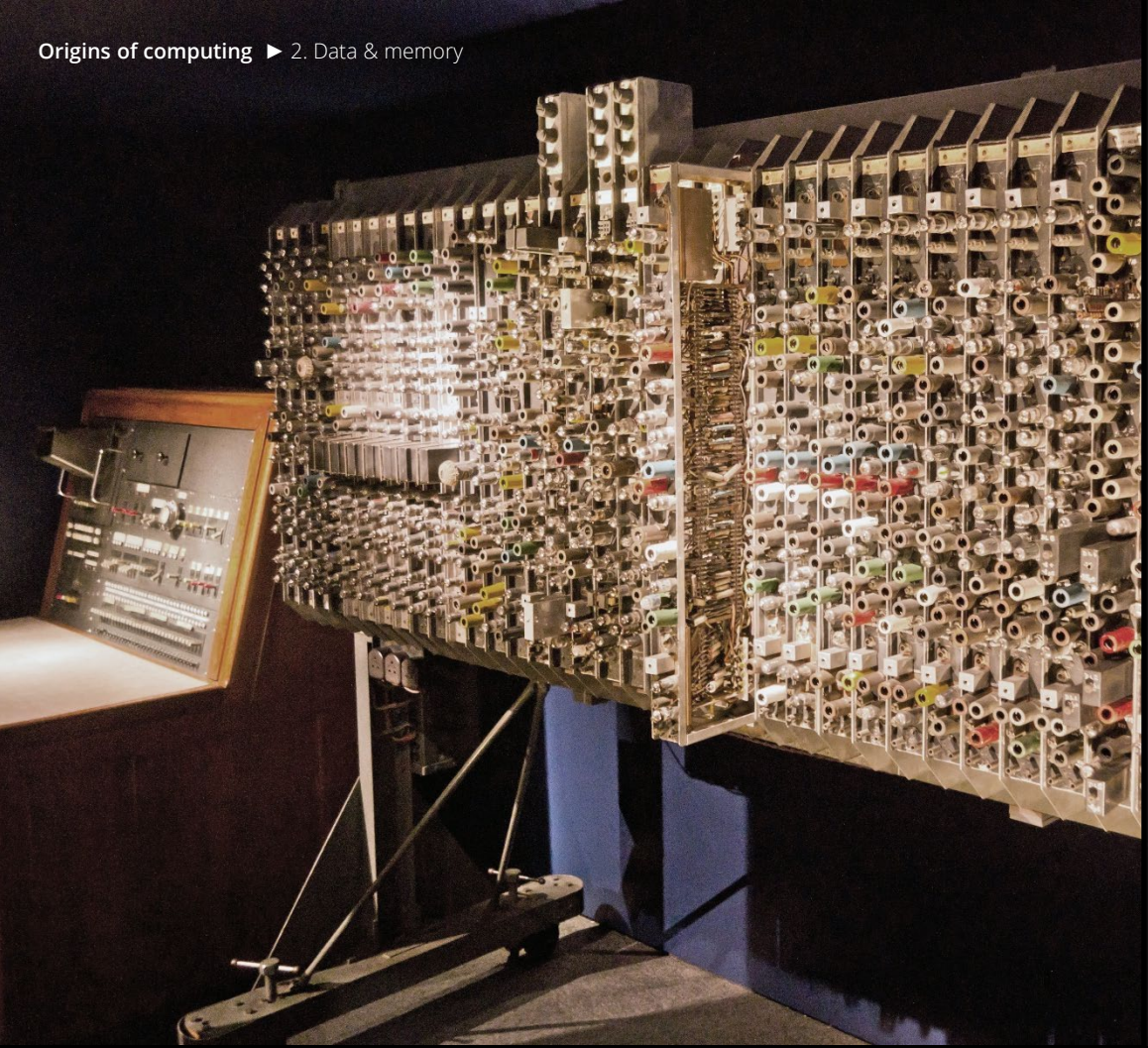
Decryption key

Reverse-algorithm



Decryption machine

Multiple reverse-engineered Enigma machines



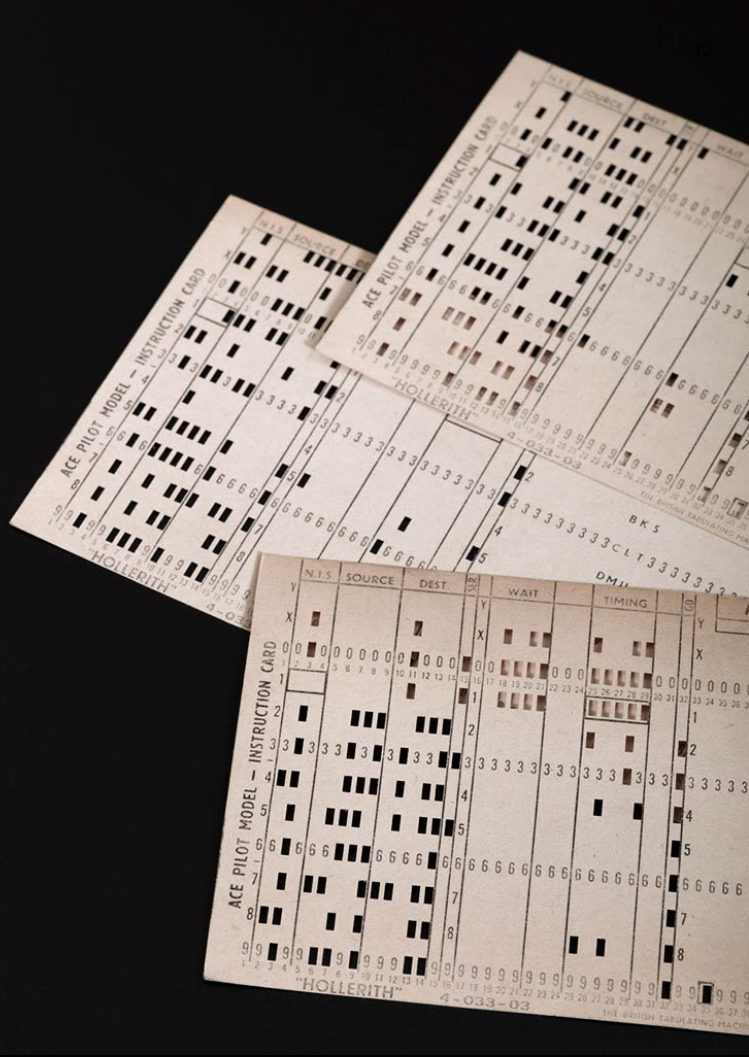


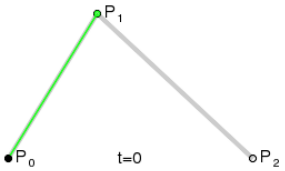
Photo: Antoine Taveneaux

Photo: Science Museum London

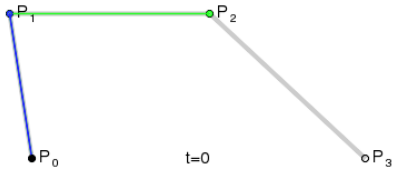
Sketchpad | Ivan Sutherland (1963)

TX-2 computer

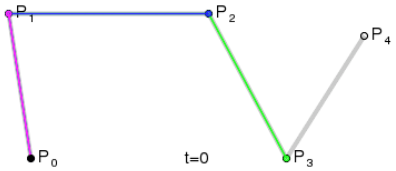
Boeing factory



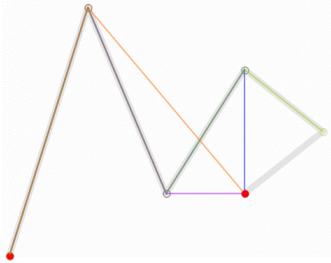
quadratic Bézier curve



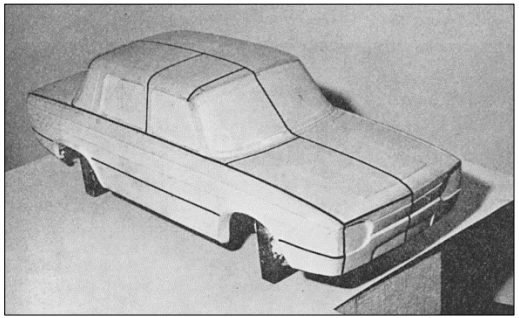
quadratic Bézier curve

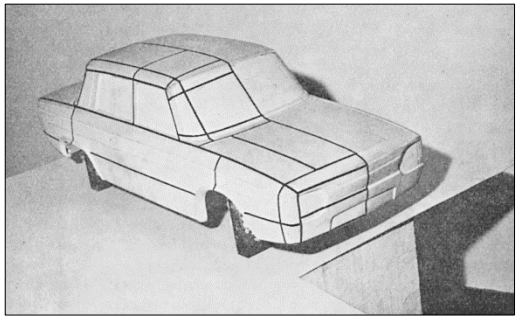


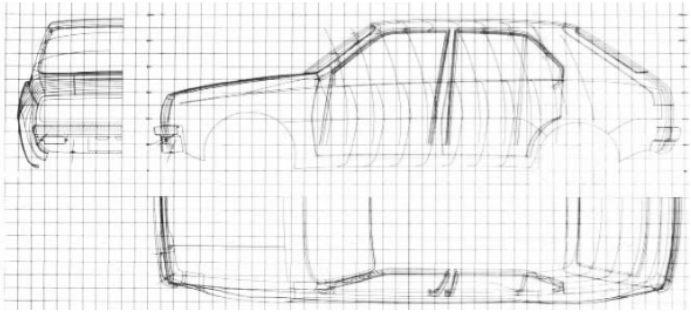
cubic Bézier curve



fifth-order Bézier curve







Bézier-curve-based UNISURZF CAD system for Renault Car Company (1960s)

Xerox Alto (1979)

First “desktop” with graphical user interface (GUI)
US$32,000 in 1979 (equivalent to US$112,726)

Epson HX-20 (1982)

first "true" laptop computer
US$795 (equivalent to US$2040.76)

Apple Lisa (1983)

Individual computing with GUI
US$9,995 (equivalent to US$25,811.18)

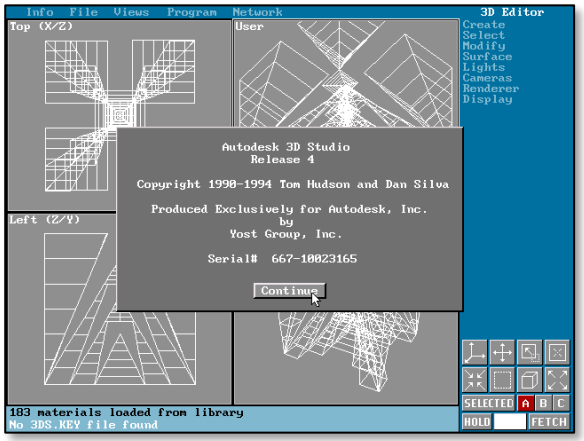
Macintosh (1984)

First “personal computer”
US$2,495 (equivalent to US$6,100)

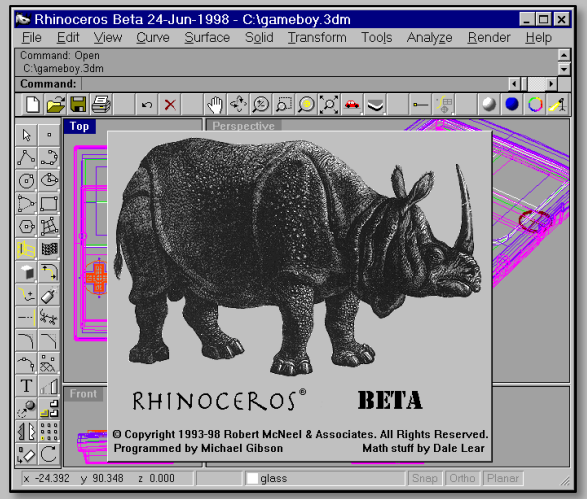
Computer-aided Design

AutoCAD v1.0 (1982)

CATIA 1 (1982)



3D Studio Max (1990)

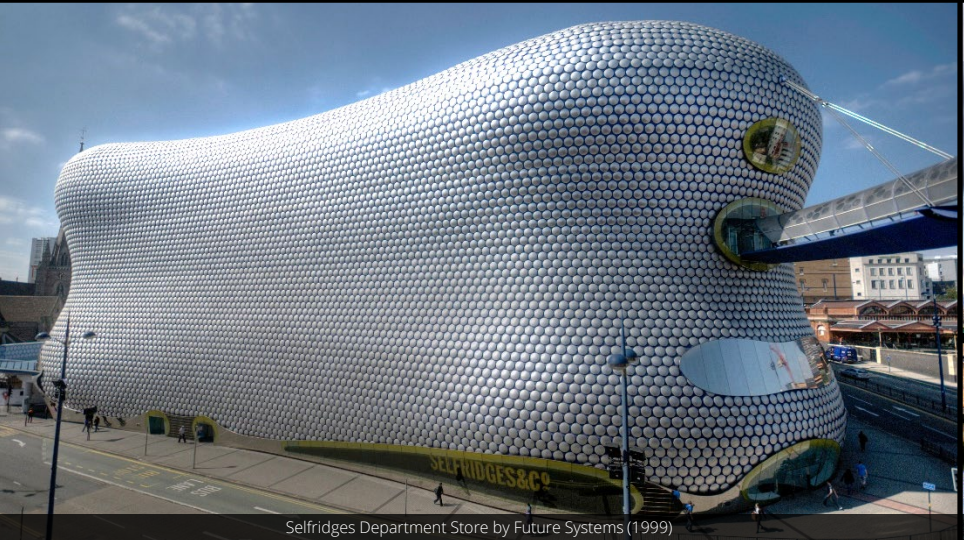


Rhinoceros/Sculptura 2 (1993)

Maya (1998)

Sage Gateshead by Foster + Partners (2004)

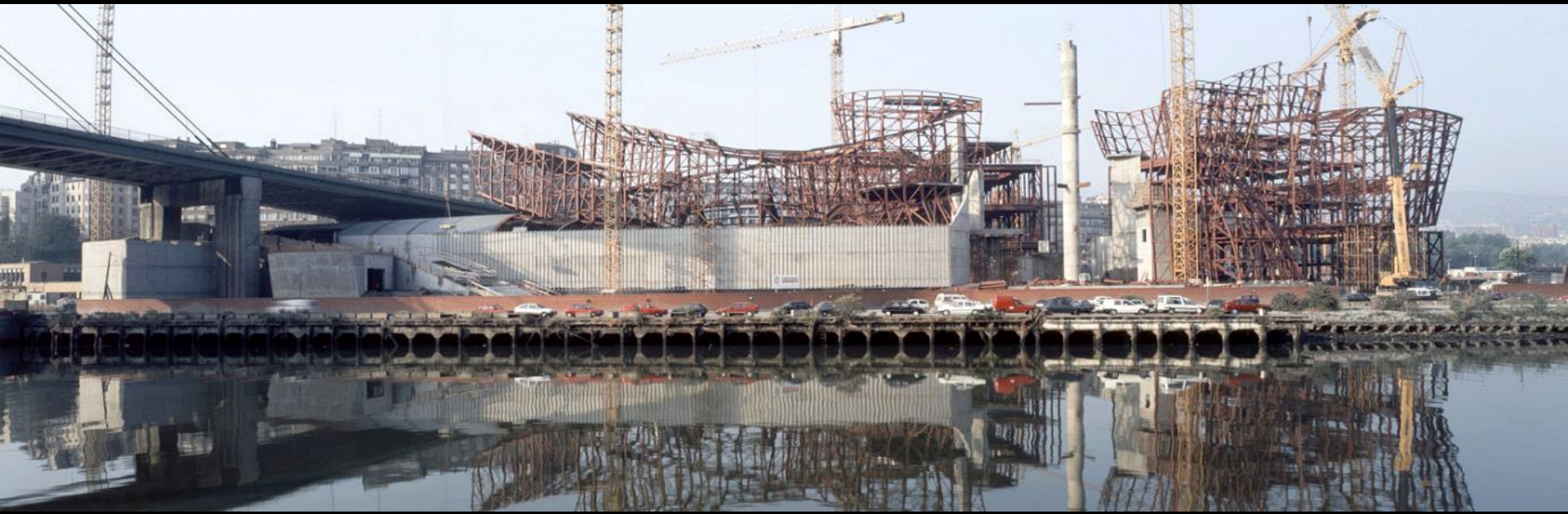
Kunsthaus Graz by Colin Fournier (2003)



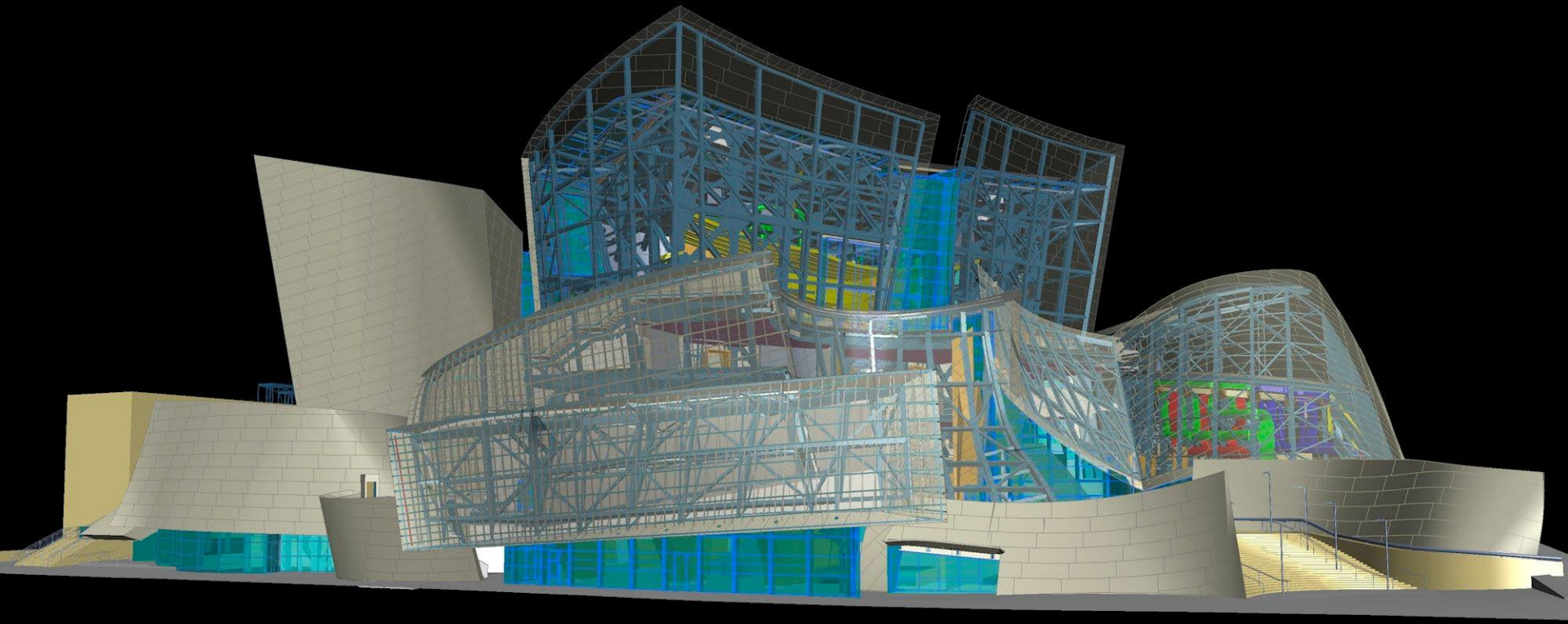
Selfridges Department Store by Future Systems (1999)

Guggenheim Museum Bilbao by Frank Gehry (1997)

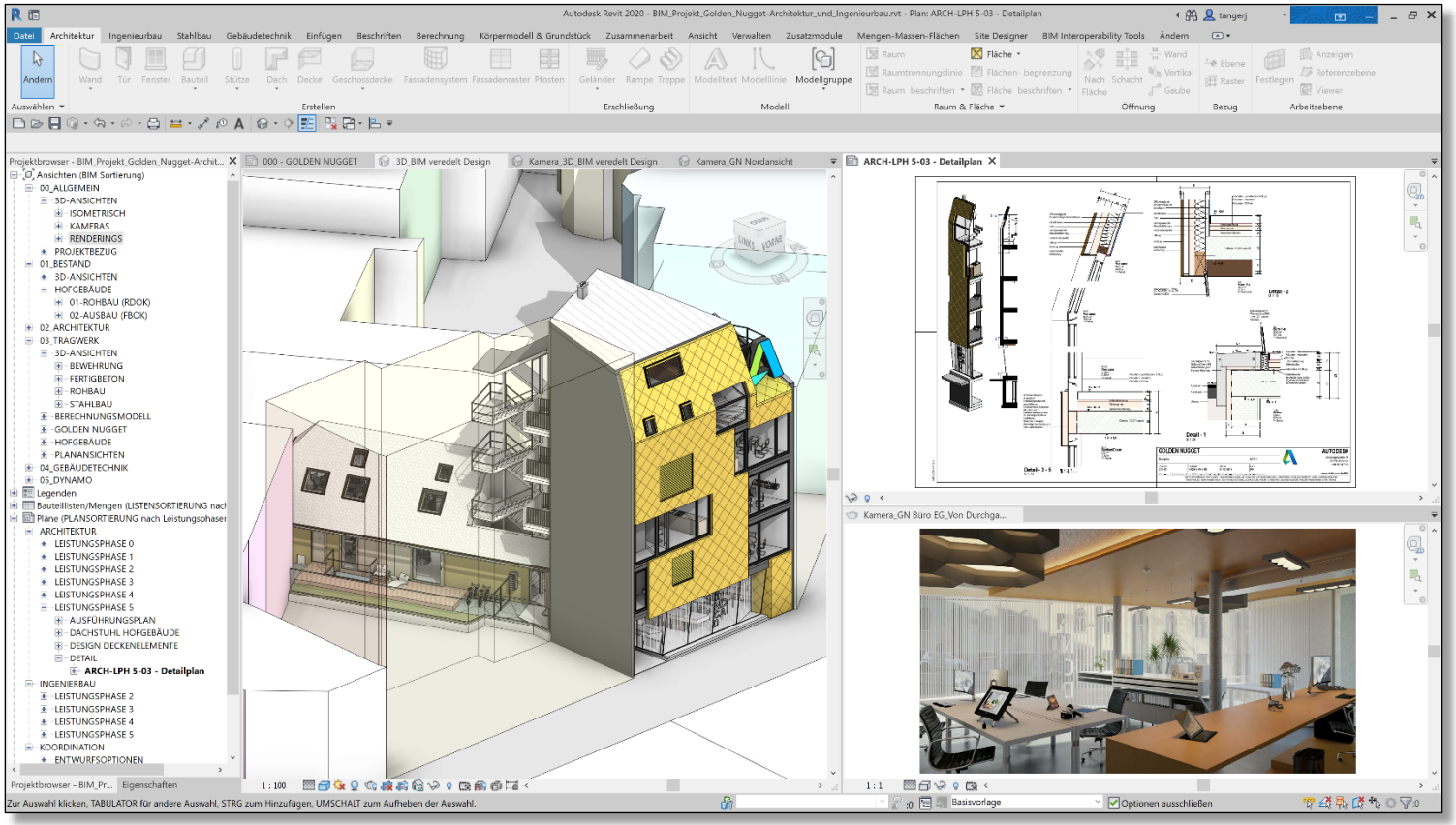

← Frank Gehry



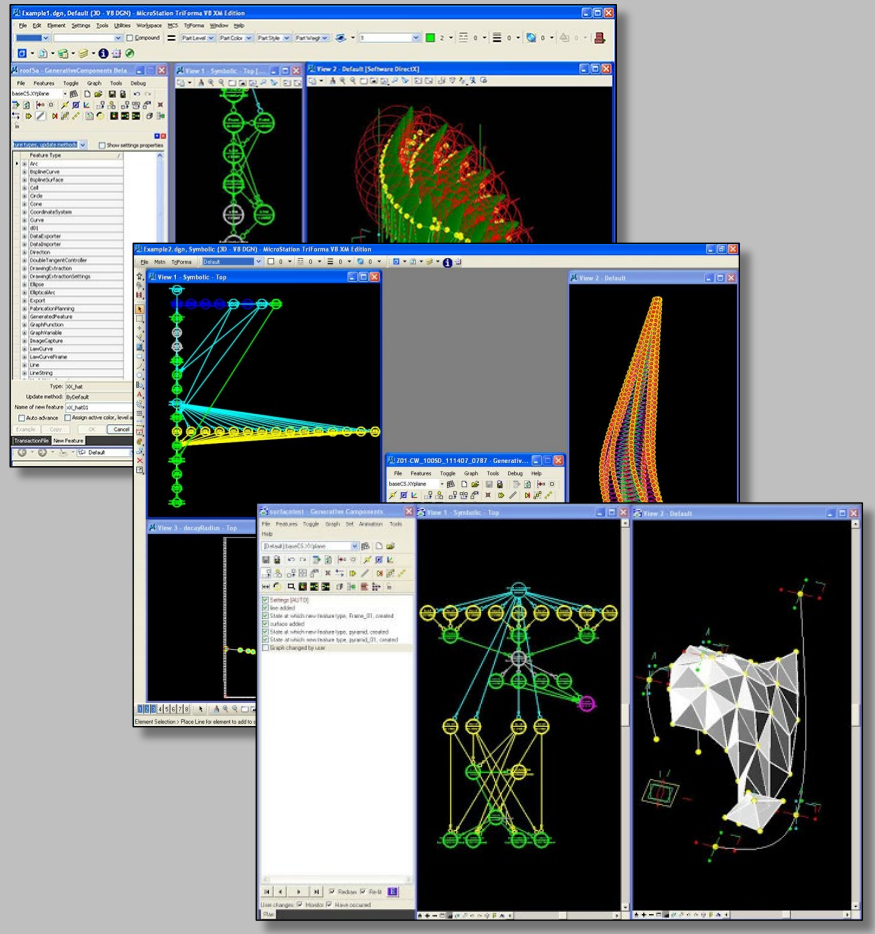
Construction of Guggenheim Museum, Bilbao, Spain, 1997 | Frank Gehry



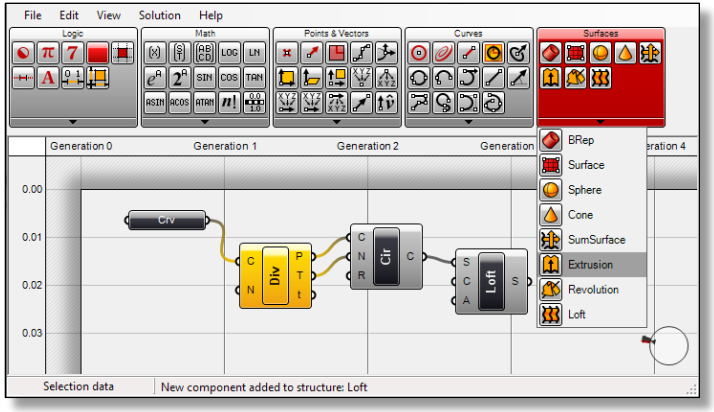
Walt Disney Concert Hall (Digital Project 3D model), Los Angeles, USA, 2003, Frank Gehry

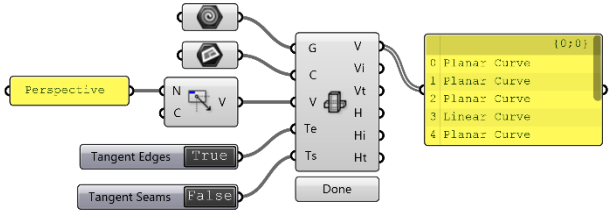

Autodesk Revit 2020 - BIM_Projekt_Golden_Nugget-Architektur_und_Ingenieurbau.rvt - Plan: ARCH-LPH 5-03 - Detailplan
Projektbrowser - BIM_Projekt_Golden_Nugget-Archit...
Ansichten (BIM Sortierung)
00_ALLGEMEIN
3D-ANSICHTEN
ISOMETRISCH
KAMERAS
RENDERINGS
PROJEKTBEZUG
01_BESTAND
02_ARCHITEKTUR
03_TRAGWERK
04_GEBÄUDETECHNIK
05_DYNAMO
Legenden
ARCHITEKTUR
LEISTUNGSPHASE 0
LEISTUNGSPHASE 1
LEISTUNGSPHASE 2
LEISTUNGSPHASE 3
LEISTUNGSPHASE 4
LEISTUNGSPHASE 5
AUSFÜHRUNGSPLAN
DACHSTUHL HOFGEBÄUDE
DESIGN DECKENELEMENTE
DETAIL
ARCH-LPH 5-03 - Detailplan
INGENIERBAU
KOORDINATION
ENTWURFSOPTIONEN
000 - GOLDEN NUGGET
3D_BIM veredelt Design
Kamera_3D_BIM veredelt Design
Kamera_GN Nordansicht
ARCH-LPH 5-03 - Detailplan
Kamera_GN Büro EG_Von Durchga...
Zur Auswahl klicken, TABULATOR für andere Auswahl, STRG zum Hinzufügen, UMSCHALT zum Aufheben der Auswahl.

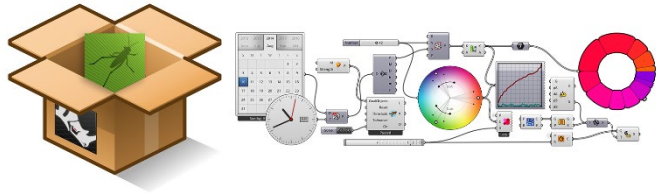
Computational ~~Computer-aided~~ Design



Generative Design (2003)



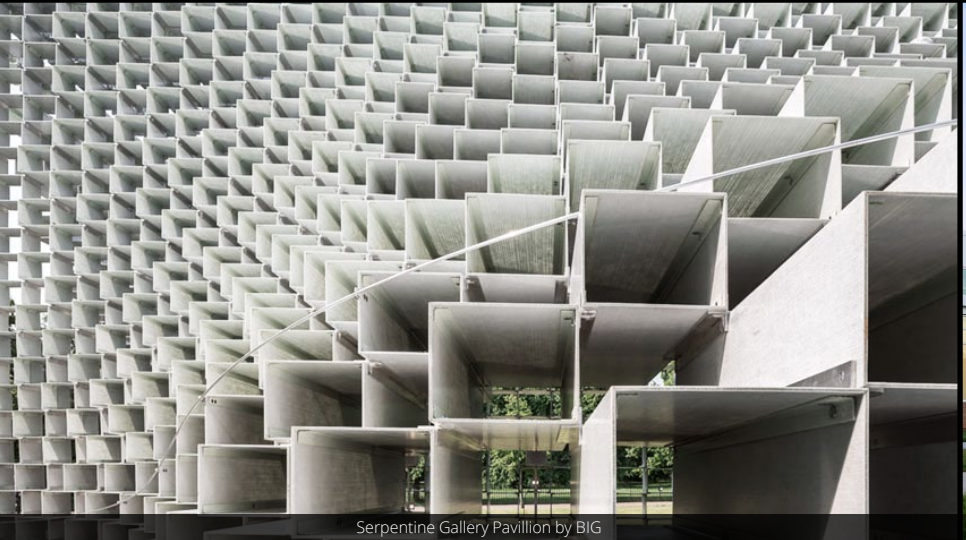




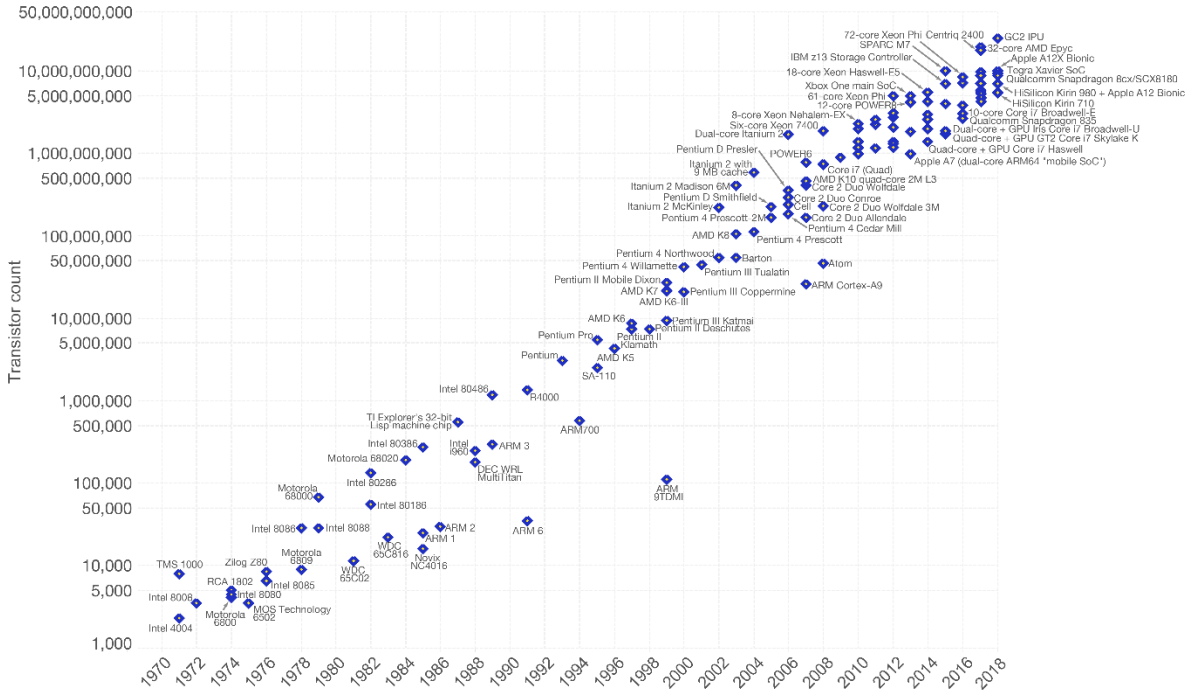
Explicit History / Grasshopper (2007)


Metropol Parasol by Jürgen Mayer


Galaxy SOHO by Zaha Hadid Architects

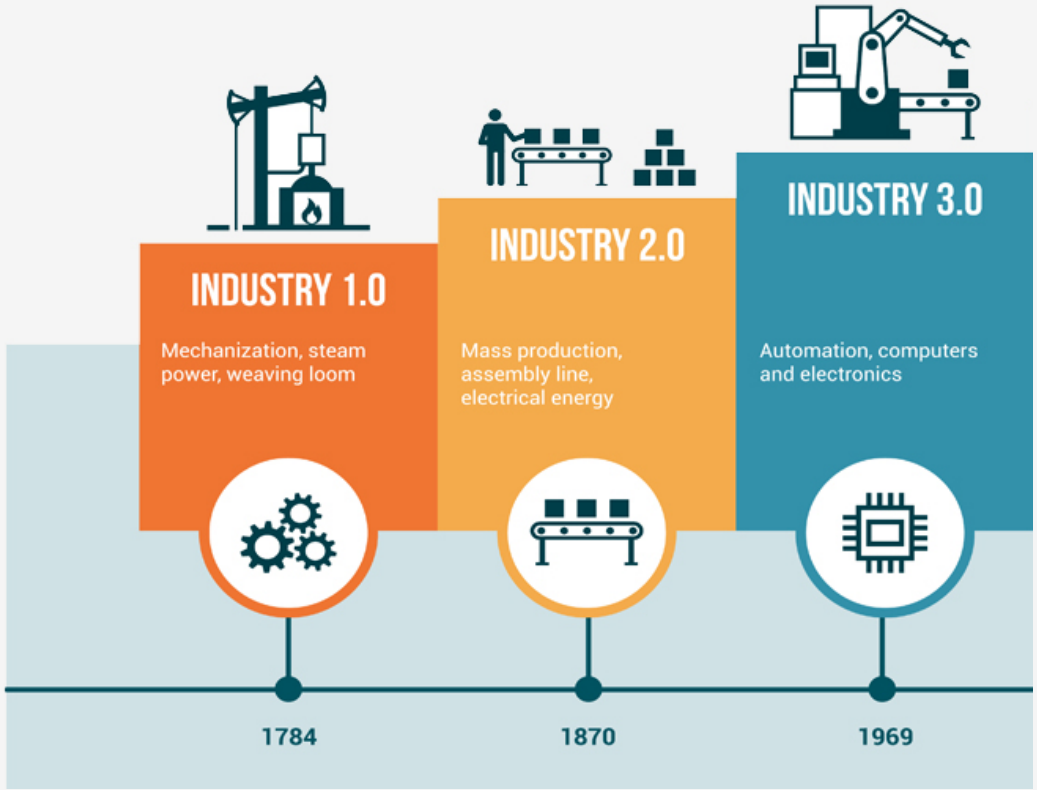

Serpentine Gallery Pavillion by BIG


Pillar of Dreams by Marc Fornes



Moore's Law – The number of transistors on integrated circuit chips (1971-2018)

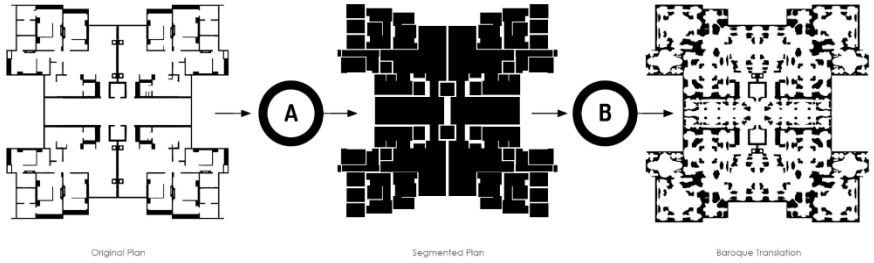
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are linked to Moore's law.


INDUSTRY 1.0
Mechanization, steam power, weaving loom
INDUSTRY 2.0
Mass production, assembly line, electrical energy
INDUSTRY 3.0
Automation, computers and electronics
1784
1870
1969

Layout + spatial configuration algorithms

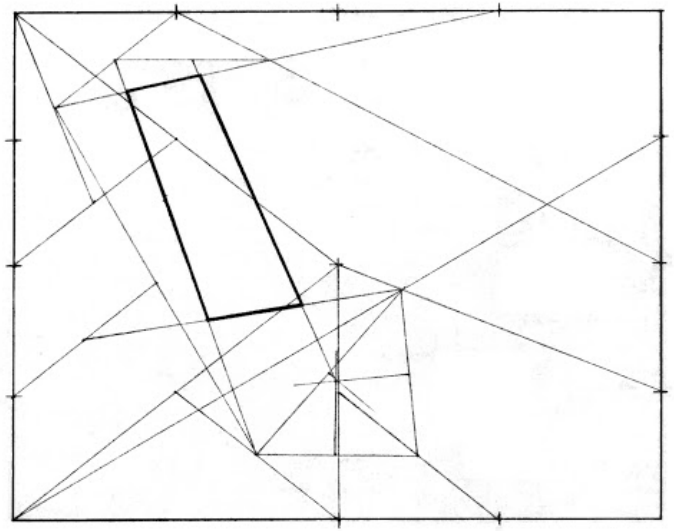
Generative design + topology optimisation

Pattern recognition + cataloguing



Style crossover

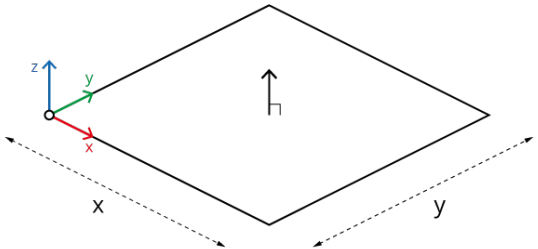
Computational Design
Algorithmic



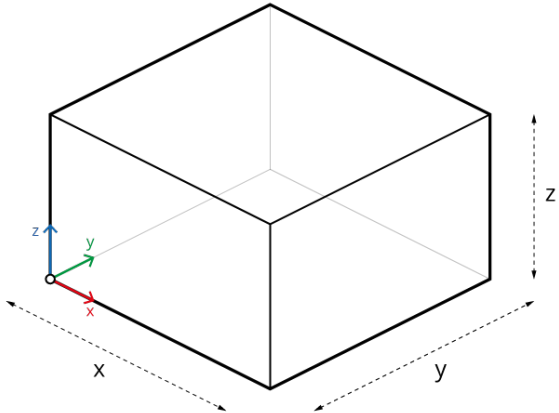
A quadrangle which is formed and enclosed by four lines, the first of which is drawn from a point halfway between a point halfway between the center of the wall and the upper left corner and the midpoint of the left side and the upper left corner to a point halfway between the midpoint of the top side and the upper right corner, the second line from a point halfway between the start of the first line and a point halfway between the midpoint of the top side and the upper left corner to a point halfway between a point halfway between the center of the wall and the lower left corner and the midpoint of the bottom side, the third line from a point halfway between a point halfway between the start of the first line and the end of the second line and a point halfway between the midpoint of the left side and the lower left corner to a point which is on an axis between the lower left corner to a point halfway between the midpoint of the right side and the upper right corner where a line drawn from the center of the wall to a point halfway between the midpoint of the right side and the lower right corner would cross that axis, the fourth line from a point equidistant from the end of the third line, the end of the second line and a point halfway between a point halfway between the center of the wall and the midpoint of the bottom side and a point halfway between the midpoint of the bottom side and the lower right corner to a point halfway between the start of the second line and a point where a line would cross the first line if it were drawn from the midpoint of the right side to a point halfway between the midpoint of the top side and the upper left corner.

– Sol LeWitt, 1974

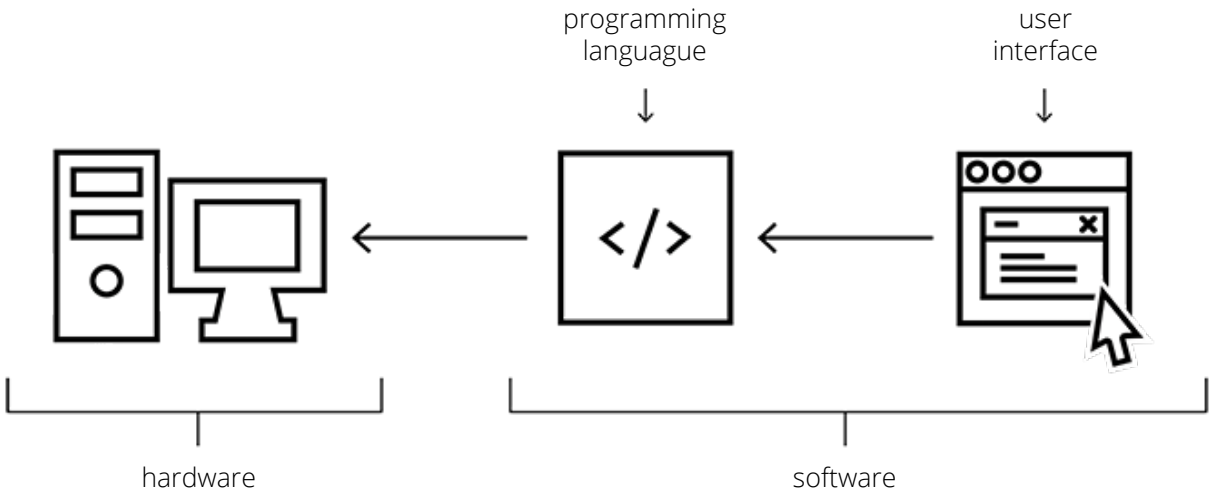
1. Choose a starting point

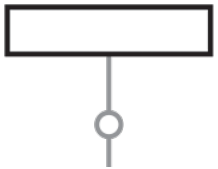


2. Draw a rectangle with dimensions X and Y on the xy plane

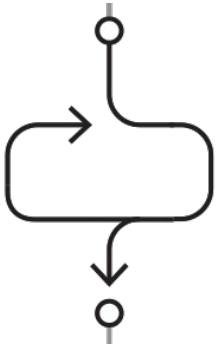


3. Extrude the rectangle in the z-axis with amount z

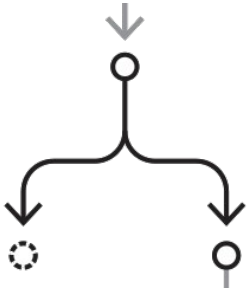

programming
languague
user
interface
</>
hardware
software



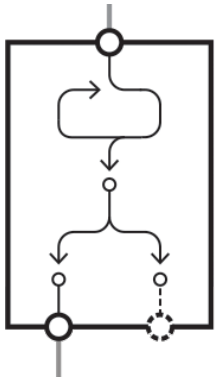
1. variables



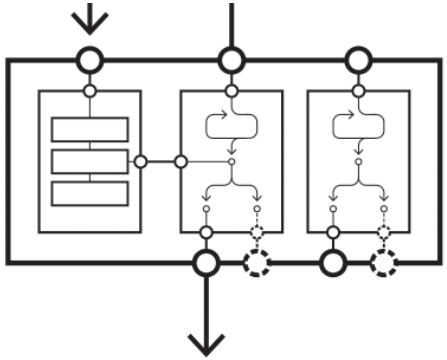
2. loops



3. conditional



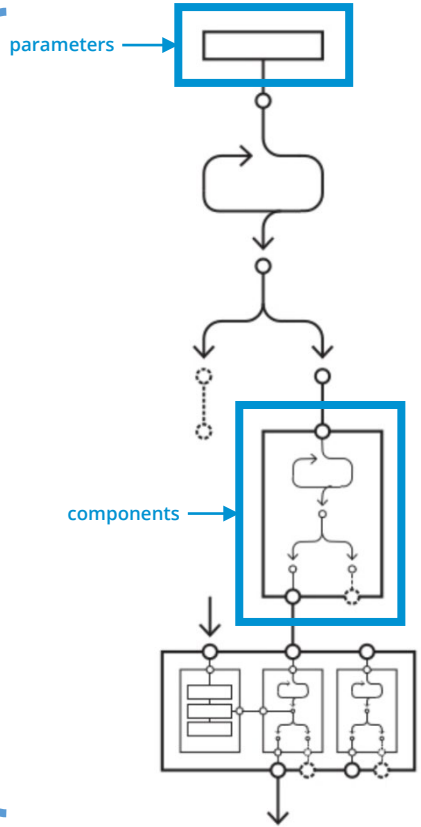
4. functions



5. objects

Adapted from Danil Nagy

visual programming
(Grasshopper)



1. variable(s)

```
x = 1
y = 'hello world'
myList = [1, 'b', 3]
myDict = {'a': 1, 'b': 2}
```

2. Loops

```
for i in range(10):
    # do something 10 times

for element in myList:
    # do something for all elements
```

3. Conditional

```
if x > 10 and y > 10:
    # do something
elif x > 5 or y < 5:
    # do something else
else:
    # do something else
```

4. Functions

```
def myFunction(input1, intput2):
    # do something
    return output

c = myFunction(a, b)
```

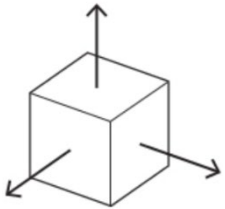
5. Object

```
class MyClass:
    def __init__(self, input1):
        self.localVar = input1
    def myMethod(self, input2):
        # do something
        return output

classInstance = MyClass(a)
c = classInstance.myMethod(b)
```

procedural programming
("scripting")

object-oriented programming
("OOP")

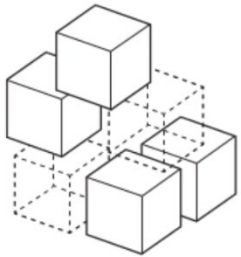


Morphological

- Continuous measures

+ Good top-down control over design

- Can usually only generate simple design spaces

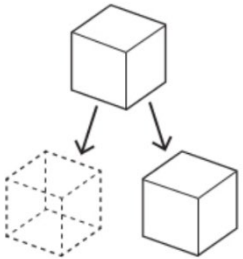


State-change

- Choices, categories

+ can create discontinuous design spaces
+ control over individual elements

- Many inputs required (each element needs to be controlled separately)

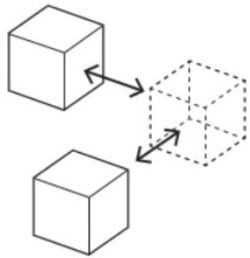


Rule-based

- L-system, shape grammars, 1D cellular automata

+ reduced number of inputs (abstraction of inputs into rule sets)
+ can create complexity

- Only top-down control
- can't control individual behaviour
- can't create emergence
- Potentially redundant or incomplete design space



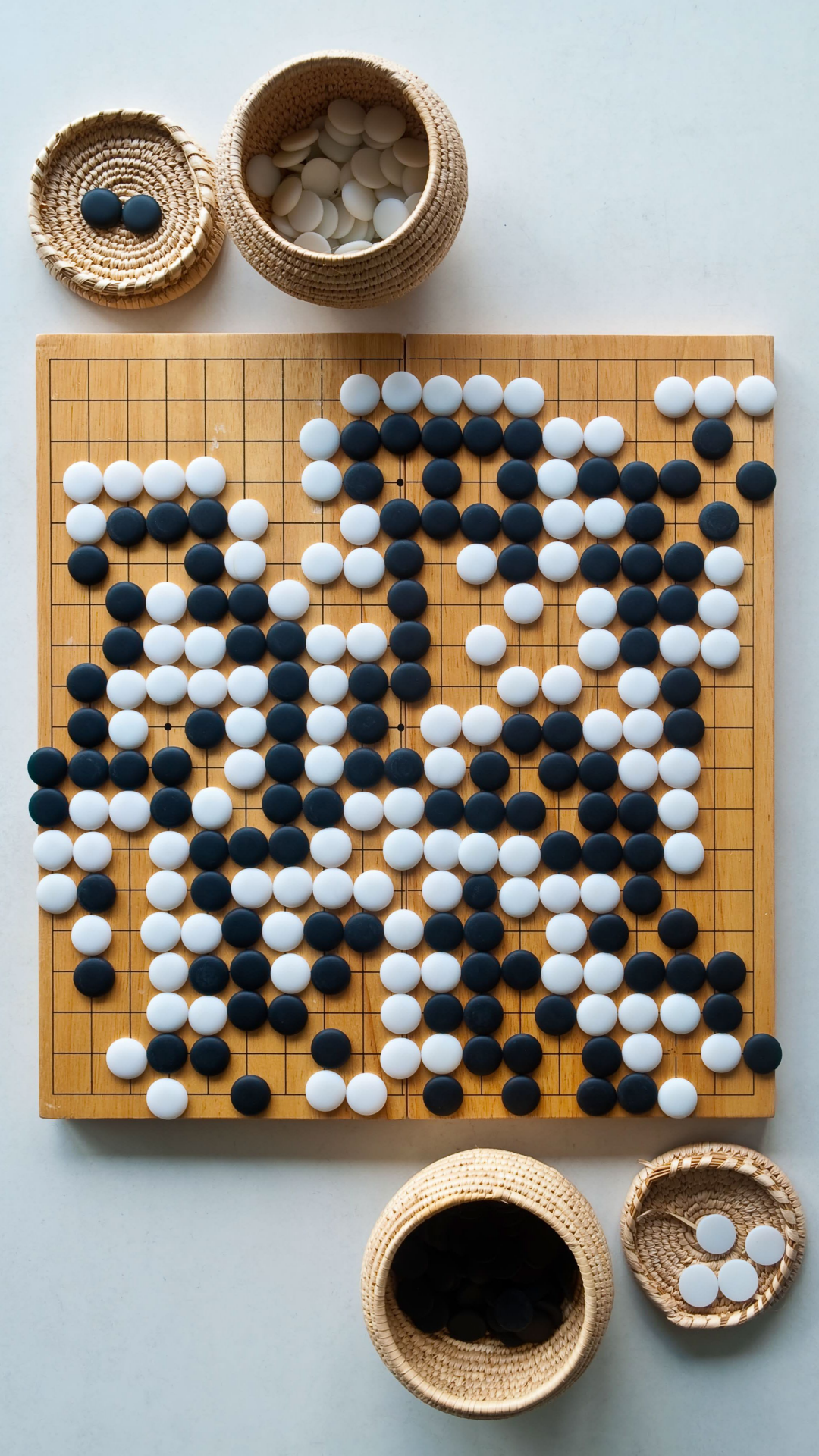
Behavioral

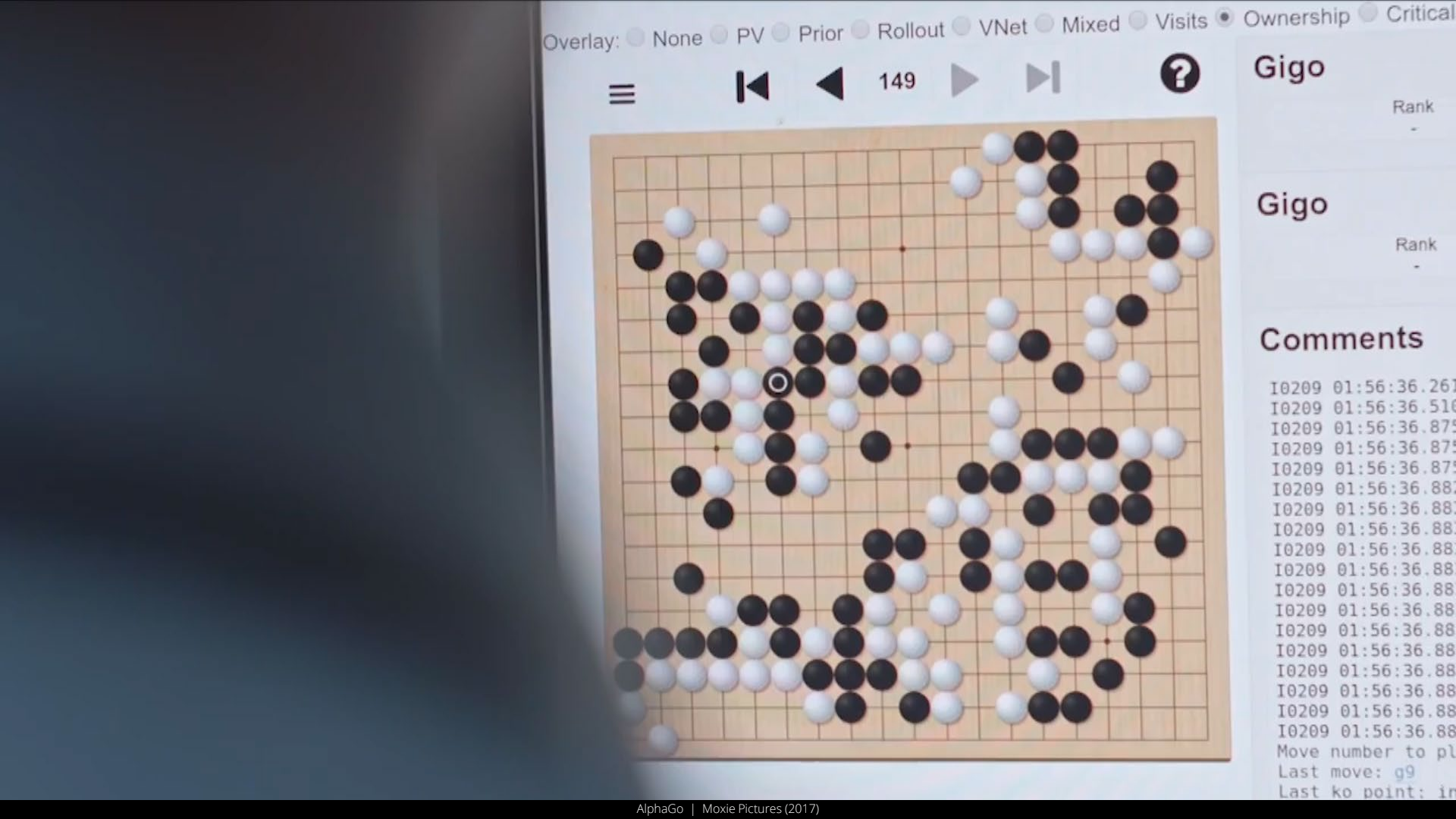
- object-oriented, agent-based behavior models (dynamic)

+ reduced number of inputs (abstraction of inputs into agent behaviors)
+ can lead to emergence

- Little intuitive control over macro design
- Potentially redundant or incomplete design space

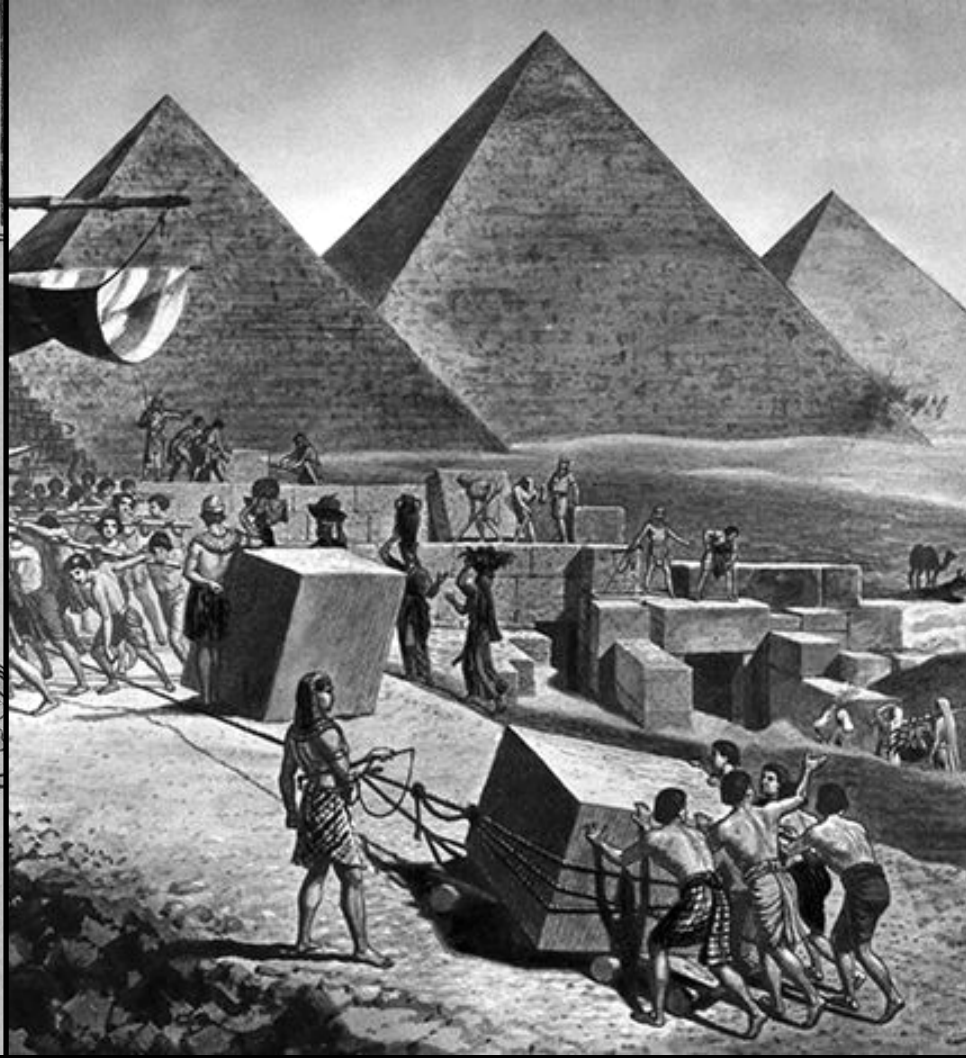
Adapted from Danil Nagy





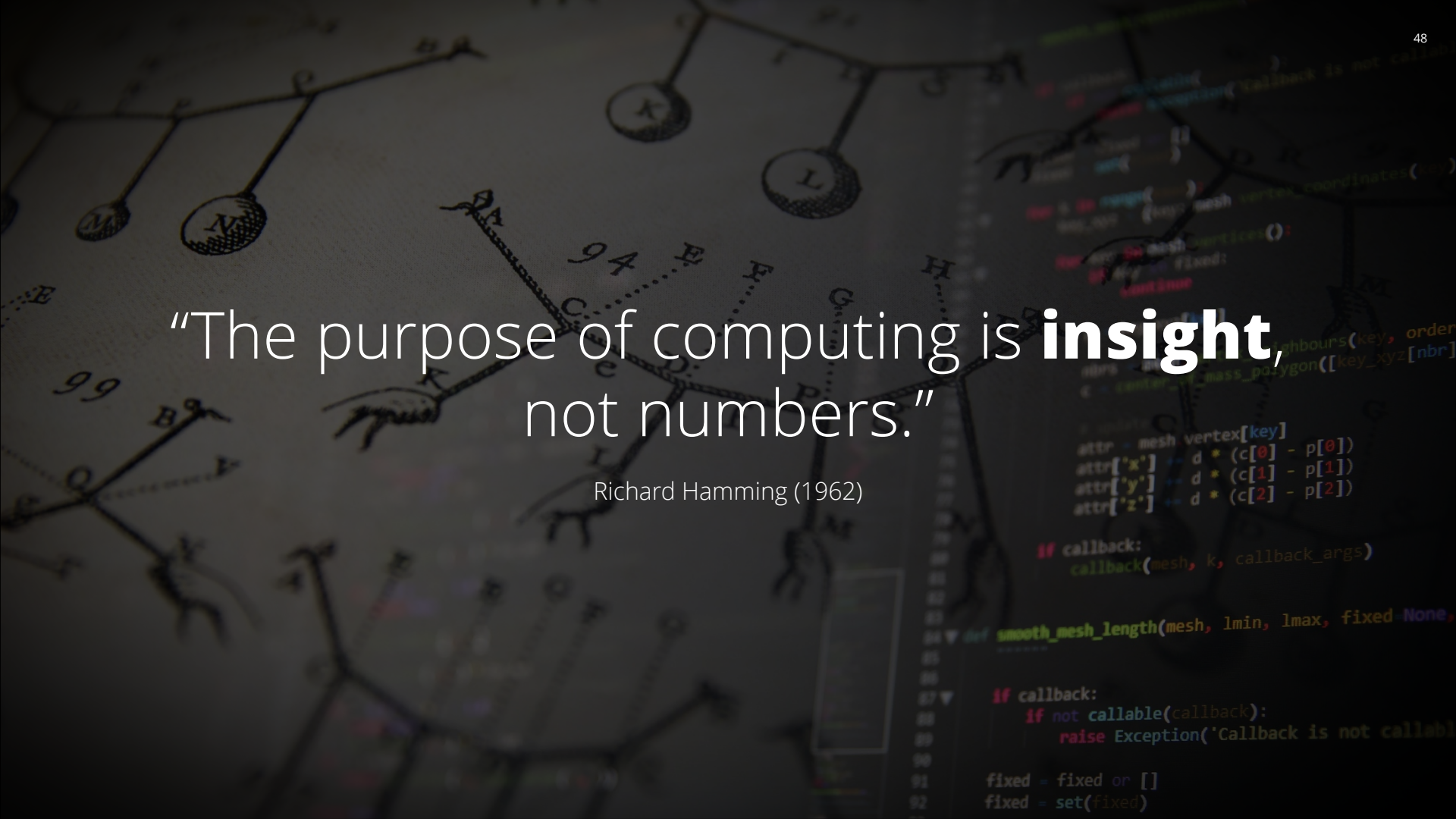
AlphaGo | Moxie Pictures (2017)

AlphaGo | Moxie Pictures (2017)




柯洁 KE JIE
00:17:0
ALPHAGO
01:51:3

37



“The purpose of computing is **insight**, not numbers.”

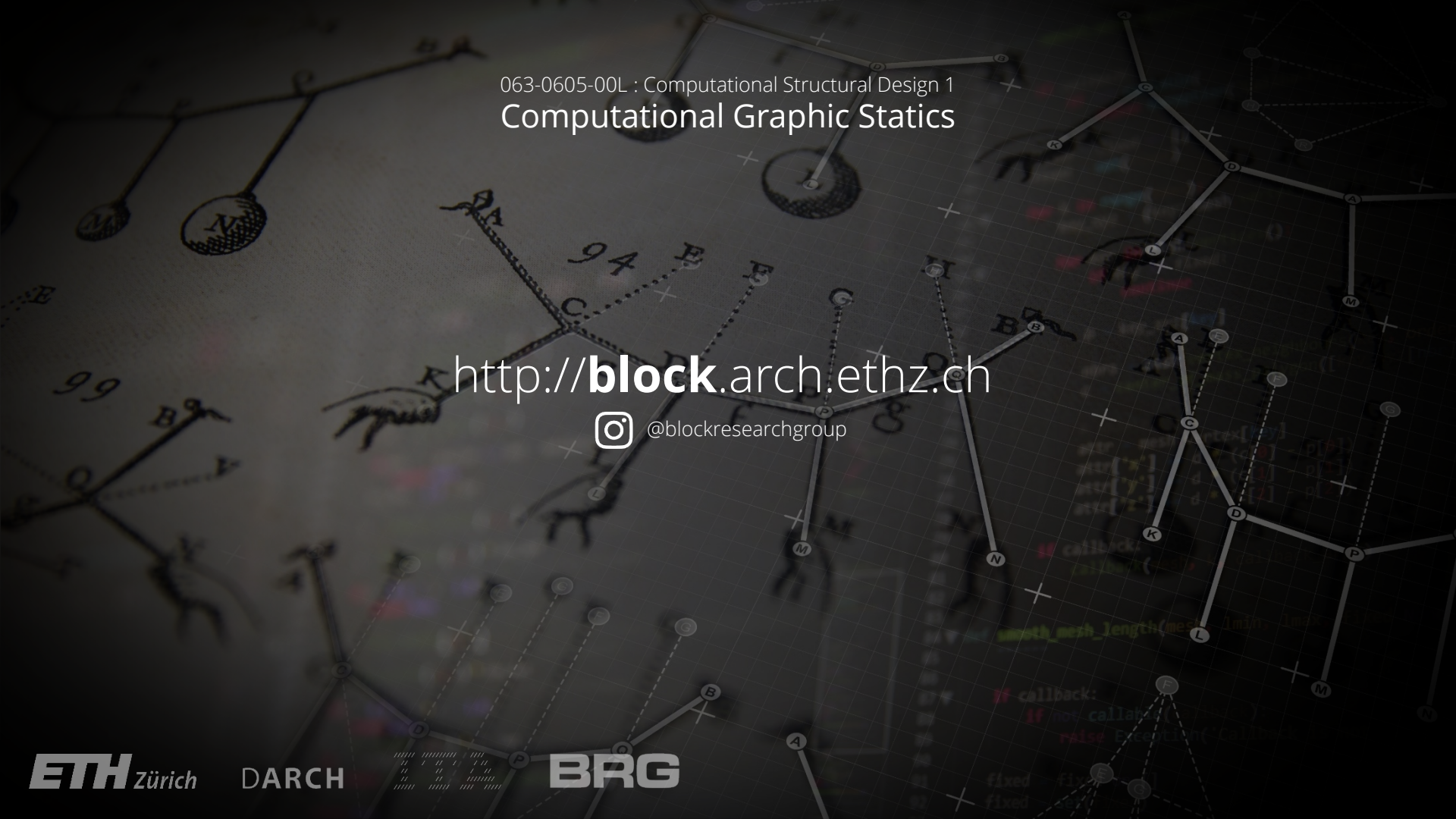
Richard Hamming (1962)

063-0605-00L : Computational Structural Design 1

Computational Graphic Statics

http://**block**.arch.ethz.ch

@blockresearchgroup



ETH Zürich DARCH BRG